

**SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL
FACULDADE DE TECNOLOGIA SENAI/SC FLORIANÓPOLIS
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

DEREK WILLIAN STAVIS

**UMA PROPOSTA DE INTERFACE DO CLIENTE DE TORRENT TRANSMISSION
BASEADO NO GTK HIG 3.14**

Florianópolis/SC

2014

DEREK WILLIAN STAVIS

**UMA PROPOSTA DE INTERFACE DO CLIENTE DE TORRENT TRANSMISSION
BASEADO NO GTK HIG 3.14**

Trabalho de Conclusão de Curso apresentado à Banca Examinadora do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas da Faculdade de Tecnologia do SENAI Florianópolis como requisito parcial para obtenção do Grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

Professor Orientador: Luciana Schmitz.

Florianópolis/SC

2014

DEREK WILLIAN STAVIS

**UMA PROPOSTA DE INTERFACE DO CLIENTE DE TORRENT TRANSMISSION
BASEADO NO GTK HIG 3.14**

Trabalho de Conclusão de Curso apresentado à Banca Examinadora do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas da Faculdade de Tecnologia do SENAI Florianópolis como requisito parcial para obtenção do Grau de Tecnólogo em Análise e Desenvolvimento de Sistemas.

**APROVADA PELA COMISSÃO EXAMINADORA
EM FLORIANÓPOLIS, 28 DE NOVEMBRO DE 2014**

Prof. Luciana Schmitz, Esp. (SENAI/SC)
Coordenador do Curso

Profa. Aline Cristina Antonelli de Oliveira, Esp. (SENAI/SC)
Coordenador de TCC

Prof. Luciana Schmitz, Dr. (SENAI/SC)
Orientador

Prof. Fulado de tal, Me. (SENAI/SC)
Examinador

“A única coisa que não muda é que tudo muda.”
(HERÁCLITO DE ÉFESO)

Resumo

STAVIS, Derek Willian. **Uma proposta de interface do cliente de torrent Transmission baseado no GTK HIG 3.14** Florianópolis, 2013. 25f. Trabalho de Conclusão de Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas. Faculdade de Tecnologia do SENAI, Florianópolis, 2013.

RESUMO

Palavras-chave: Interfaces Gráficas. GTK+. GNOME. Design. Redesign.

Os avanços trazidos nos últimos quatro anos no ambiente gráfico GNOME foram de suma importância para atingir uma experiência diferenciada e consistente de uso. Conceitos já difundidos no design de interfaces de usuário foram revisitados, paradigmas foram quebrados, mudando a forma com que os usuários vêem e interagem com o sistema operacional. Entretanto, nem todos os softwares disponíveis para a plataforma acompanharam a velocidade de desenvolvimento da plataforma GNOME, e muitos ainda necessitam de atenção.

Com o lançamento do HIG (Guia de Interface com o Usuário) no GNOME versão 3.12 toda filosofia e padrões de design da plataforma foram sintetizados em linguagem simples e objetiva, tornando-se o material referência para o desenvolvimento e manutenção de seu ecossistema de softwares. Esta pesquisa aborda o processo de adaptação do cliente de torrent Transmission de acordo com o guia de interfaces do GNOME versão 3.14.

STAVIS, Derek Willian. **Uma proposta de interface do cliente de torrent Transmission baseado no GTK HIG 3.14** Florianópolis, 2013. 89f. Trabalho de Conclusão de Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas. Faculdade de Tecnologia do SENAI, Florianópolis, 2013.

ABSTRACT

This is the english abstract.

Key-words: Graphical User Interfaces. GTK+. GNOME. Design. Redesign.

LISTA DE ILUSTRAÇÕES

| | |
|---|-----------|
| Figura 1 – Uma captura de tela do GNOME 3.16 | 13 |
| Figura 2 – Padrões e suas aplicações | 16 |
| Figura 3 – Transmission 2.28 no GNOME 3.16 | 17 |
| Figura 4 – Transmission no GNOME 3.16 | 21 |

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

| | |
|-------|--|
| GNU | GNU is Not Unix - Projeto de licenciamento e distribuição de pacotes de software, baseados na licença GNU GPL. |
| GPL | General Public License - Modelo de licenciamento de software baseado em liberdades. |
| GTK | GIMP Toolkit - Biblioteca de componentes para criação de interfaces gráficas. |
| API | Application-Programmer Interface - Protocolos de comunicação de um determinado pacote de software. |
| GIMP | GNU Image Manipulation tool - Ferramenta de manipulações de imagem de código aberto, sob licença GNU. |
| GNOME | Ambiente gráfico de estação de trabalho disponível para GNU/Linux. |
| HIG | Human Interface Guidelines - Guia de referências de design para implementação de interfaces. |

SUMÁRIO

| | |
|---|-----------|
| 1 INTRODUÇÃO | 10 |
| 1.1 OBJETIVOS | 10 |
| 1.1.1 Objetivo geral | 10 |
| 1.1.2 Objetivos específicos | 10 |
| 2 REVISÃO DA LITERATURA | 11 |
| 2.1 Interfaces gráficas com o usuário | 11 |
| 2.2 Gerenciadores de Janelas | 12 |
| 2.3 Toolkits Gráficos | 12 |
| 2.4 O toolkit GTK e a plataforma GNOME | 13 |
| 2.5 Software livre e o desenvolvimento contínuo | 14 |
| 2.6 Impactos na consistência e usabilidade | 14 |
| 2.7 HIG ou Human Interface Guidelines | 15 |
| 2.8 Estudo de caso: Transmission | 15 |
| 3 PROCEDIMENTOS METODOLÓGICOS | 18 |
| 3.1 Análise cronológica dos padrões de design nas aplicações | 18 |
| 3.2 Identificação de pontos de redesign na janela principal do Transmission | 19 |
| 3.3 Proposição de melhorias na interface do Transmission | 20 |
| 4 RESULTADOS E DISCUSSÕES | 21 |
| 4.1 A evolução das aplicações centrais analisadas | 21 |
| 4.2 Barra de Título | 21 |
| 4.3 Barra de Filtros | 22 |
| 4.4 Limites Globais de Upload/Download | 22 |
| 4.5 Botão Limite de velocidade | 22 |
| 4.6 Estísticas da barra de status | 22 |
| 5 CONCLUSÃO | 23 |
| REFERÊNCIAS | 24 |

1 INTRODUÇÃO

O lançamento de um guia de padrões para o desenvolvimento de interfaces gráficas orientadas ao ambiente gráfico GNOME foi de suma importância para alinhar desenvolvedores ao intuito deste ambiente gráfico de código aberto.

O guia contribui para que o GNOME passe a ter mais harmonia visual, seja através dos aplicativos centrais, – os primeiros a receber tratamento estético – ou por aplicativos de terceiros, que devem se adaptar utilizando o guia como referência.

Aplicativos que não se adequam ao guia passam a apresentar-se defasados, perdendo pontos em usabilidade e em beleza, aumentando a taxa de rejeição. Mantenedores de aplicativos baseados no toolkit GTK devem, portanto, dedicar atenção para o fator usabilidade, mantendo-se compatível com o progresso da plataforma GNOME.

Um aplicativo que sofre deste efeito e necessita dedicação é o Transmission, um cliente de torrent popular, que foi desenvolvido na era GTK 2, onde os requisitos, recursos e a visão da plataforma GNOME eram diferentes (BENSON; CLARK; NICKELL, 2014).

O objetivo principal desta pesquisa é explorar o HIG do GTK 3.14, além de avançar o desenvolvimento de um software livre mantido pela comunidade de código aberto, propondo e contribuindo melhorias visuais e de código-fonte na interface gráfica do Transmission.

Esta pesquisa foi baseada na análise de diversos casos de uso de softwares contidos na plataforma GNOME Desktop, versão 3.14, e tem a pretensão de descrever o processo de pensamento e a motivação por trás das adaptações de interface gráficas propostas.

1.1 OBJETIVOS

1.1.1 Objetivo geral

Propor adaptações na interface gráfica GTK do cliente de torrent Transmission de acordo com as recomendações do HIG versão 3.14, trazendo mais harmonia para seus utilizadores na plataforma GNOME.

1.1.2 Objetivos específicos

1. Comparar dados evolutivos da interface de aplicações centrais do GNOME.
2. Identificar pontos de redesign na janela principal do Transmission.
3. Propor adaptações na interface do Transmission, embasadas no HIG 3.14.

2 REVISÃO DA LITERATURA

O desenvolvimento de mecanismos que automatizam tarefas existe desde a idade das pedras, onde hominídeos produziam ferramentas para auxílio próprio. A transversalidade do conhecimento e da experimentação nos levou a descoberta de novas metodologias e o aperfeiçoamento de técnicas, consequentemente modificando a forma com que o ser humano interage com o mundo (SARTORI, 2010, p. 1).

2.1 Interfaces gráficas com o usuário

O conceito de interface é extremamente amplo, e foi largamente difundido com o início dos estudos de interação humano-máquina. O avanço da industrialização permitiu com que máquinas extremamente complexas substituíssem seres humanos nas tarefas mais difíceis, deixando estes, seus operadores, apenas com a responsabilidade de pilotá-las de forma simples e segura.

O advento de tecnologias multimídias interativas como computadores, celulares e tablets elevou o patamar da criação de interfaces e gestão de tarefas ao estado da arte, agregando conhecimentos transversais de artistas visuais a músicos.

Ferreira, Barr e Noble (2005, p. 1) afirmam que a criação de “interfaces com o usuário ainda está mais para arte do que para ciência.”, e justificam, explicando que “A maior parte do design ou redesign é baseado em estudos empíricos ou protótipos, e ainda há muito pouco compreensão teórica ou de engenharia de como conduzir o processo de design e produzir bons designs pela primeira vez”.

Interfaces gráficas modernas foram alcançadas através da associação entre hardware e software, podendo utilizar de um ou mais dispositivos de entrada e saída, com o intuito de promover usabilidade e fácil adaptação.

Dispositivos de entrada transformam coordenadas do mundo real para o mundo virtual, registrando condições externas que podem ser modificadas através da interação com um ou mais atores, os usuários. Dispositivos de entrada comuns são mouse, tela de toque, teclado, etc.

Dispositivos de saída projetam informações geradas por um sistema computacional e seu caráter é geralmente baseado nos sentidos: Visão, audição, tato. O dispositivo de saída mais comum é o monitor, que tem por finalidade projetar imagens compatíveis com a capacidade humana de visão.

2.2 Gerenciadores de Janelas

Gerenciadores de janelas tem como principal função dividir a imagem de um dispositivo de saída (geralmente monitores) em múltiplas regiões de desenho, denominadas janelas (MYERS, 1996, p. 5).

O primeiro conceito acadêmico de gerenciamento de tarefas através da sobreposição de janelas data de 1969, na tese de Ph.D. de Alan Kay. A implementação de seu conceito, vista pela primeira vez em funcionamento no sistema do Xerox PARC, é largamente utilizada até hoje por grandes sistemas tanto comerciais quanto de código aberto, como Windows, OS X, GNOME, KDE (MYERS; HUDSON; PAUSCH, 2000, p. 7).

Um gerenciador de janelas moderno também tem por finalidade coordenar a exibição de um conjunto de janelas em um conjunto de monitores, escutar por eventos de entrada (mouse, teclado) e informar os responsáveis pelas janelas sobre alterações no layout de tela (dimensões da tela, dimensões da janela, espaço de cor).

Gerenciadores de janelas, porém, não tem por responsabilidade preencher as áreas de desenho com gráficos, e como suas APIs operam geralmente a nível de pixel, a tarefa de escrever um programa gráfico acaba sendo demorada e entediante. Além disso, se cada desenvolvedor criasse seus próprios componentes, seria praticamente impossível disponibilizar uma experiência consistente ao usuário Rosenthal (1988 apud MYERS; HUDSON; PAUSCH, 2000).

Para solucionar este problema ferramentas conhecidas como toolkits foram criadas sobre as abstrações disponibilizadas pelos WMs.

2.3 Toolkits Gráficos

Do ponto de vista do programador a finalidade de um toolkit é abstrair características de baixo nível, disponibilizando uma fachada homogênea e portável, além de comportamento e experiência visual consistente para o usuário final (MYERS; HUDSON; PAUSCH, 2000).

Do ponto de vista financeiro o reuso de código-fonte é uma maneira de amortizar o custo de desenvolvimento, reduzindo o tempo de desenvolvimento de novos projetos Haefliger, Krogh e Spaeth (2008).

As responsabilidades de um toolkit incluem desenhar elementos de interface gráfica interativos, como texto, botões, imagens, indicadores de progresso, etc, de acordo com um ou mais estilos visuais. Estes elementos de interface são chamados de componentes, ou widgets.

Também é sua responsabilidade processar eventos de um ou mais dispositivos de entrada (mouse, teclado, painel de toque) verificar a colisão de um evento com um widget (clique em um botão, por exemplo) e por fim traduzir e informar os eventos para a aplicação proprietária da janela.

Toolkits multiplataforma podem ser utilizados para escrever interfaces gráficas portáveis, permitindo com que o mesmo código seja recompilado para um sistema operacional diferente do em que foi escrito e funcione da mesma forma.

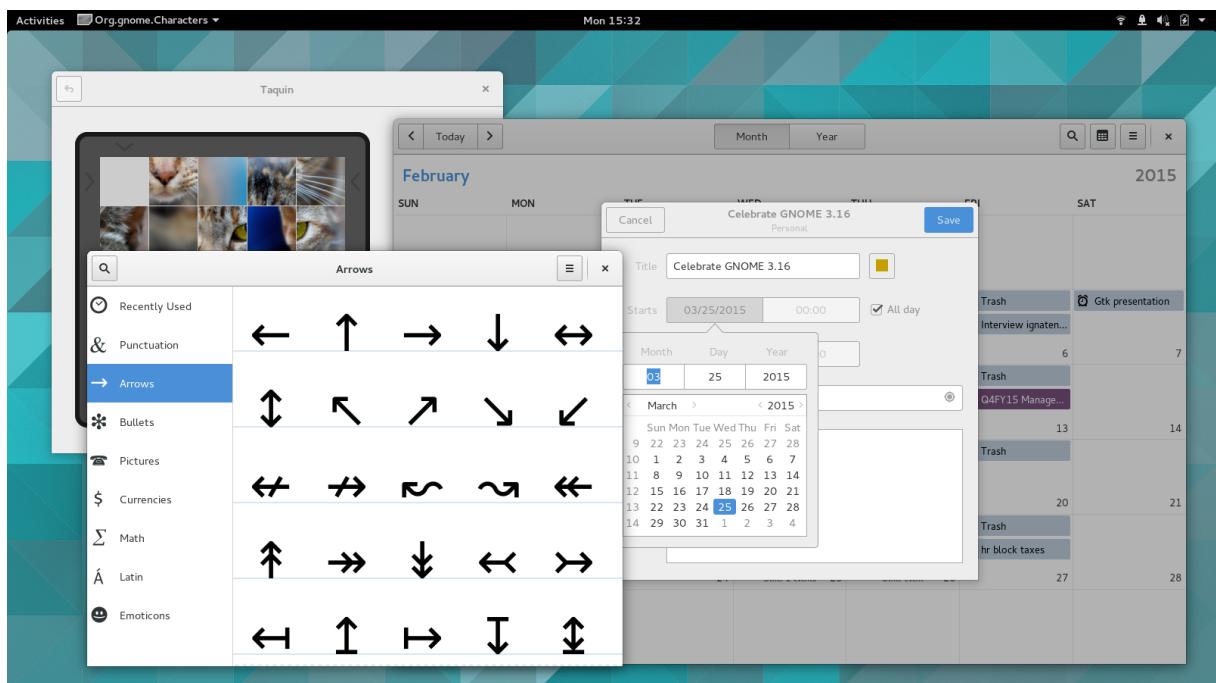
2.4 O toolkit GTK e a plataforma GNOME

Existem diversas opções de gerenciadores de janela de código aberto, comumente incluídos em distribuições de Linux. O GNOME, um ambiente gráfico bastante difundido pelos usuários de Linux, foi fundado e está em ativo desenvolvimento por uma comunidade de engenheiros de software ao redor do mundo.

O GNOME 3 é uma maneira fácil e elegante de usar seu computador. Ele foi desenhado para lhe colocar no controle e trazer liberdade para todos. O GNOME 3 é desenvolvido pela comunidade GNOME, um grupo internacional e diverso de contribuidores que são suportados por uma fundação independente, e sem fins lucrativos (THE GNOME PROJECT, 2015).

Muito mais que um gerenciador de janelas, a experiência GNOME é construída por um conjunto de aplicações centrais, que incluem um gerenciador de janelas, um lançador de aplicações e diversos aplicativos centrais, como calculadora, editor de texto, gerenciadores de arquivos, redes, contatos, etc.

Figura 1 – Uma captura de tela do GNOME 3.16



Fonte: Matias Clasen

A plataforma GNOME e seus aplicativos centrais são baseados no toolkit GTK. O GTK + é formado por um conjunto de ferramentas multi-plataforma para criar interfaces gráficas de

usuário. Por oferecer um conjunto completo de widgets é adequado para projetos desde pequenas ferramentas pontuais até suítes completas de aplicativos (THE GTK+ PROJECT, 2015).

2.5 Software livre e o desenvolvimento contínuo

Uma das característica das plataformas de código aberto é a distribuição de esforços em prol do constante desenvolvimento e melhoria. A pluralidade de opiniões e idéias eleva o patamar das discussões e permite com que vários pontos de vista sejam levados em consideração na evolução da plataforma.

Apesar dos prós existentes na distribuição de esforços também existem os contras – Projetos que são desenvolvidos paralelamente nem sempre avançam na mesma velocidade. O contra fica mais sério quando um projeto depende do outro, como é o caso de toolkits e programas que consomem suas APIs.

Quando ocorrem mudanças na interface de programação de uma framework ou biblioteca, softwares dependentes tem de se adaptar as mudanças. O efeito cascata provocado pela propagação de alterações na malha de softwares dependentes não é incomum, e já foi objeto de estudo (YAU; COLLOFELLO; MACGREGOR, 1978).

2.6 Impactos na consistência e usabilidade

Além dos aplicativos centrais, geralmente garantidos de acompanhar a evolução da plataforma – sua ergonomia e visual – existe uma vasta gama de aplicações tanto de código aberto quanto proprietárias disponíveis para atender as mais variadas necessidades.

A grande maioria das aplicações GTK+ são mantidas pela comunidade de software livre, e se não atualizadas podem ficar defasadas em usabilidade, ergonomia e consistência com a plataforma. Shneiderman (1992) descreve a usabilidade como uma combinação das seguintes características:

- Facilidade de aprendizado
- Alta velocidade de operação
- Baixa taxa de erros
- Satisfação do usuário
- Retenção de usuários pelo tempo

Em prol da usabilidade a maioria dos toolkits aplica um modelo próprio de apresentação, organização, interação e estilização de widgets, atendendo aos requisitos do tipo de ambiente onde é executado (desktop, tablet e celular).

2.7 HIG ou Human Interface Guidelines

Divulgado no ano de 2014, acompanhando a versão 3.14 da plataforma GNOME, sob o título de Human Interface Guidelines, o conjunto de padrões de design oficialmente promovido pela GNOME Foundation busca promover a máxima integração de interfaces gráficas GTK+ na plataforma GNOME.

Se você é um desenvolvedor com experiência limitada de design o HIG foi planejado para auxiliar você a criar facilmente uma interface de usuário efetiva. Para designers, o HIG provê uma introdução as possibilidades ao usar o GTK+, assim como padrões de design que são usados nos aplicativos GNOME (DAY, 2014a).

O HIG, como também é chamado, é uma literatura ilustrada de diretrizes recomendadas no desenvolvimento de interfaces gráficas que utilizem o toolkit, com o intuito de reforçar a consistência visual e integração com o ambiente GNOME. A figura 2 ilustra alguns dos padrões de design documentados pelo HIG.

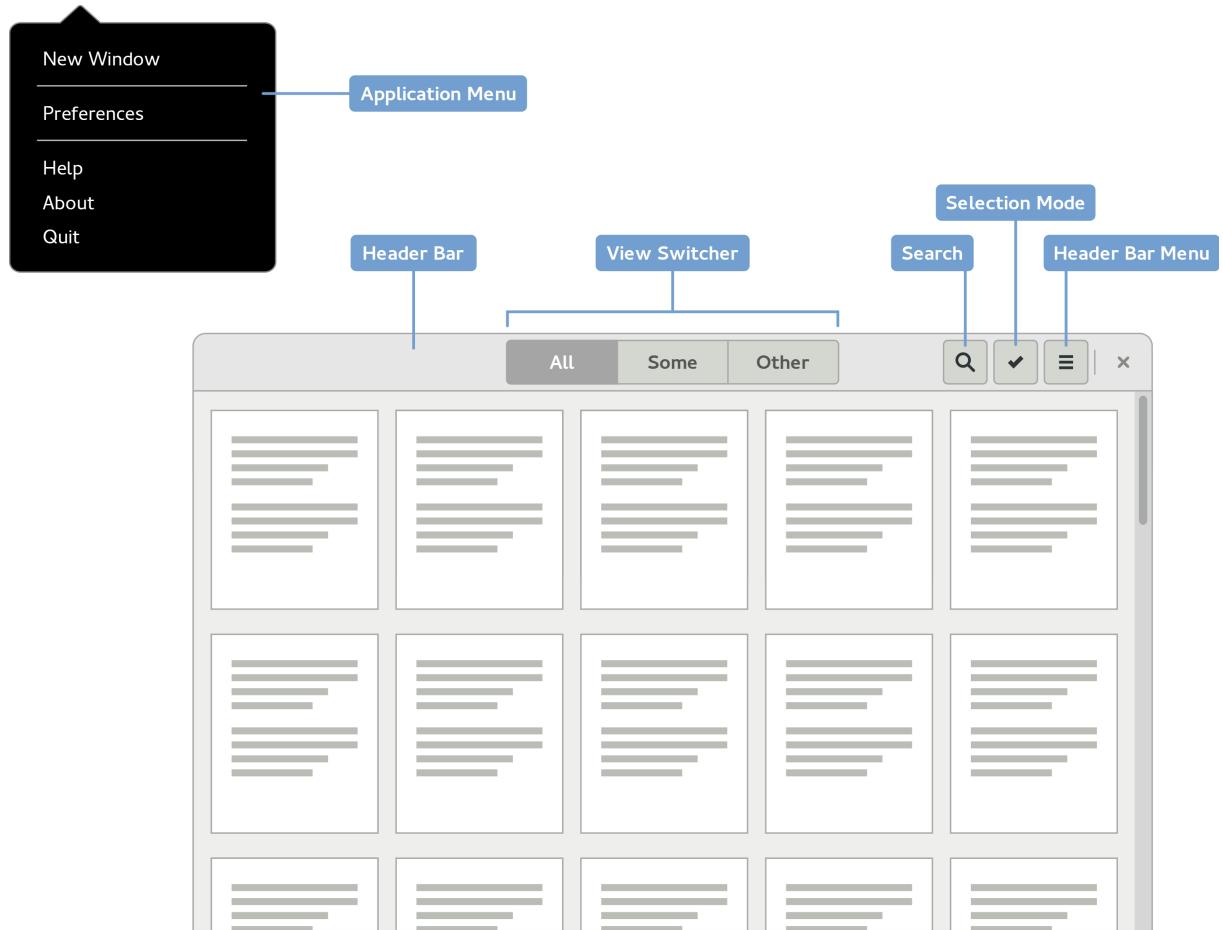
Dentre os objetivos do HIG estão a transmissão das metas de design de alto nível e estratégicas da experiência GNOME e a comunicação das diretrizes de design essenciais, de uma forma clara, e viva, acompanhando a evolução da plataforma ??).

2.8 Estudo de caso: Transmission

Um aplicativo famoso disponível para a plataforma GNOME é o cliente de BitTorrent Transmission (The Transmission Project,). Aclamado pela sua simplicidade, o aplicativo é utilizado para compartilhar arquivos através da internet.

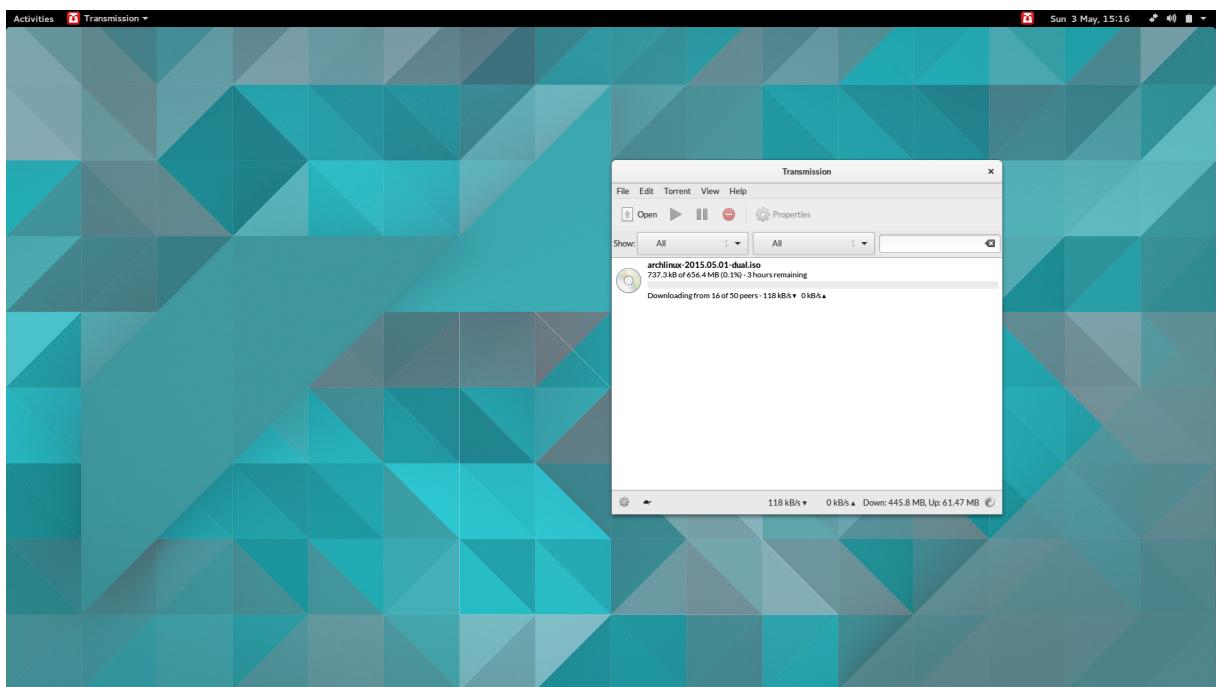
Utilizando-se de uma configuração padrão funcional e poucos cliques para configurar funcionalidades avançadas a utilização do cliente de torrent Transmission foi projetada para ser fácil e poderosa (THE TRANSMISSION PROJECT, 2015).

A primeira versão do Transmission foi lançada em Setembro de 2005, já com uma interface gráfica baseada em GTK+, projetada sob as recomendações do HIG da época, publicada digitalmente em formato de livro (BENSON; CLARK; NICKELL, 2014).

Figura 2 – Padrões e suas aplicações

Fonte: GNOME 3.16 HIG - Patterns

Figura 3 – Transmission 2.28 no GNOME 3.16



Fonte: Do autor

3 PROCEDIMENTOS METODOLÓGICOS

Em um primeiro instante, com o objetivo de avaliar o processo de evolução da plataforma GNOME e seu design, foram identificadas aplicações centrais do GNOME 3.16 que já implementam os padrões de design definidos no HIG.

Pelo importante papel desempenhado na experiência GNOME, foram selecionadas três aplicações centrais:

- Nautilus – Navegador de Arquivos
- Evince – Visualizador de Documentos Digitais
- Gedit – Editor de Arquivos de Texto

3.1 Análise cronológica dos padrões de design nas aplicações

Uma análise cronológica foi elaborada através da comparação de duas versões distantes do GNOME e das aplicações centrais escolhidas. Os parâmetros de análise foram escolhidos dentro dos padrões de design especificados pelo HIG (DAY, 2014b):

1. Menu da Aplicação
2. Janela Primária
3. Barra de Título
4. Comutador de Visão
5. Busca

O processo de análise cronológica foi facilitado pela disponibilidade de imagens oficiais de um sistema operacional com todo software necessário para uma experiência GNOME básica.

Detalhadas na tabela ?? estão as informações das versões escolhidas, dentre as disponíveis para análise (DOWNLOAD..., 2015).

Popper (1959) afirma que no ramo da pesquisa experimental é de suma importância a reproduzibilidade de um experimento, argumentando inclusive que ocorrências não reproduzíveis não tem significância para a ciência.

A soma de verificação de cada arquivo foi utilizada para checar a integridade do sistema operacional, e todas aplicações foram analisadas na versão oficial, sem alterações na configuração do sistema, troca de temas, fontes, etc.

| Versão do GNOME | Data de Modificação | Nome do arquivo | Soma de Verificação |
|-----------------|---------------------|-----------------------|--|
| 3.6.0 | 08/10/2012 | GNOME-3.6.0.iso | MD5 753c99ce2342f658 65c1f74bc3722e44 |
| 3.16 | 25/03/2015 | gnome-3.16.x86-64.iso | SHA256 4a6185a0aca89f15 8f769d76d5a0086f 0f1e9d709a5d80cd cf2b0d52d67ab2b2 |

A execução das imagens se deu em ambiente virtual, provisionado utilizando o software Oracle VM VirtualBox 4.3.26 com a seguinte configuração de máquina:

| Máquina | |
|---------------------|-----------------------------------|
| Memória Base | 4096MB |
| Processadores | 2 |
| Aceleração | VT-x/AMD-V, Nested Paging, PAE/NX |
| Vídeo | |
| Memória | 128MB |
| Aceleração | 3D |
| Sistema Operacional | |
| Tipo | Linux |
| Versão | Fedora |

O registro gráfico das aplicações analisadas se deu através de captura de telas, quais podem ser encontradas na íntegra no apêndice ??.

Pelas diferenças existentes na análise de caráter cronológico, os padrões de design escolhidos foram relacionados nas duas versões analisadas, transportando, quando necessário, a forma e função de um padrão presente mais antigo com o padrão analisado.

3.2 Identificação de pontos de redesign na janela principal do Transmission

Utilizando-se do mesmo ambiente de testes e procedimentos de aquisição de dados adquiridos na seção 3.1 foram feitos registros gráficos do Transmission na versão mais recente do GNOME. Migrações de padrões de design identificados foram transportados para o contexto do Transmission.

3.3 Proposição de melhorias na interface do Transmission

Com os resultados prévios desta pesquisa e o HIG em mãos foram elencados pontos de retrabalho na interface do Transmission. A documentação oficial e fatores como utilização prévia por ambos aplicativos centrais e outras versões do Transmission foram levados em consideração em sua formulação.

A prototipação de interfaces é uma técnica de evolução de design que tem como base a constante interação e avaliação dos resultados (BEER et al., 2009). Bäumer et al. (1996, p. 2) separa em quatro classes os protótipos de interfaces gráficas:

1. De apresentação
2. Funcional
3. Experimental
4. Piloto

A utilização de protótipos funcionais permite implementar estrategicamente ambas interface gráfica e funcionalidade de um programa para averiguar o funcionamento de um conceito.

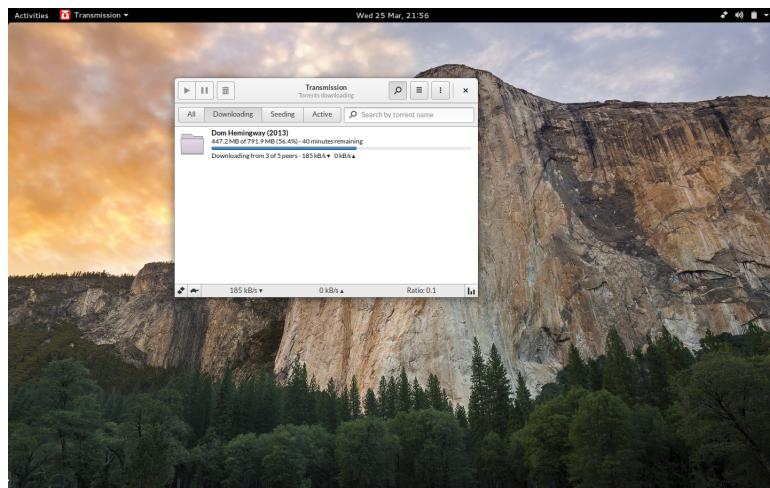
Sendo o Transmission um software de código aberto, licenciado nos termos da GPL, seu código fonte é divulgado publicamente e pode ser alterado para fins de estudo.

Com os pontos de retrabalho em mãos foi produzido um protótipo funcional para averiguar a validade das propostas e sua integração na plataforma GNOME.

4 RESULTADOS E DISCUSSÕES

O cenário ilustrado na 4 evidencia que a adaptação proposta na interface do Transmission provê maior harmonia com outros aplicativos da plataforma GNOME.

Figura 4 – Transmission no GNOME 3.16



4.1 A evolução das aplicações centrais analisadas

A análise das aplicações centrais do GNOME trouxe resultados efetivos sobre a transição gráfica e funcional dos principais widgets utilizados pela aplicação.

Um ítem notável foi a redução na utilização de espaço vertical em aplicações centrais. Day (2015) analisa a redução do uso vertical de espaço em várias versões do Nautilus e apresenta a seguinte tabela:

| Versão do Nautilus | Moldura Vertical |
|--------------------|------------------|
| 2.22 | 171 pixels |
| 3.0 | 116 pixels |
| 3.6 | 70 pixels |
| 3.12 | 48 pixels |

Observou-se que o fenômeno se repete em todas as aplicações centrais analisadas:

4.2 Barra de Título

Um dos elementos de interface mais notáveis é a barra de título. Até então desenvolvedores tinham pouco ou nenhum controle sobre elas por questões de compatibilidade de software.

Com o surgimento do conceito de ‘Header Bars’, as barras de título ganharam botões, caixas de texto, sliders, etc.

4.3 Barra de Filtros

Mostrar todos os controles de uma só vez torna a aplicação mais difícil de usar. Portanto é necessário

4.4 Limites Globais de Upload/Download

4.5 Botão Limite de velocidade

4.6 Estísticas da barra de status

5 CONCLUSÃO

REFERÊNCIAS

- BÄUMER, D. et al. User interface prototyping—concepts, tools, and experience. In: IEEE COMPUTER SOCIETY. *Proceedings of the 18th international conference on Software engineering*. [S.I.], 1996. p. 532–541. Citado na página 20.
- BEER, D. J. D. et al. Client-centred design evolution via functional prototyping. *International Journal of Product Development*, Inderscience, v. 8, n. 1, p. 22–41, 2009. Citado na página 20.
- BENSON, C.; CLARK, B.; NICKELL, S. *GNOME Human Interface Guidelines*. 2014. GNOME Human Interface Guidelines 2.2.1. Disponível em: <<https://developer.gnome.org/hig-book/2.32>>. Acesso em: 5 apr. 2015. Citado 2 vezes nas páginas 10 e 15.
- DAY, A. *GNOME Human Interface Guidelines*. 2014. GNOME Human Interface Guidelines. Disponível em: <<https://developer.gnome.org/hig/3.14>>. Acesso em: 5 apr. 2015. Citado na página 15.
- DAY, A. *Patterns*. 2014. GNOME Human Interface Guidelines. Disponível em: <<https://developer.gnome.org/hig/3.14>>. Acesso em: 5 apr. 2015. Citado na página 18.
- DAY, A. *GNOME design: saving space since 2009 (or so)*. 2015. GNOME design: saving space since 2009 (or so). Disponível em: <<https://blogs.gnome.org/aday/2014/08/27/gnome-design-saving-you-space-since-2009-or-so/>>. Acesso em: 3 may. 2015. Citado na página 21.
- DOWNLOAD.GNOME.ORG. 2015. Disponível em: <<https://download.gnome.org/misc/promo-usb/>>. Acesso em: 25 may. 2015. Citado na página 18.
- FERREIRA, J.; BARR, P.; NOBLE, J. The semiotics of user interface redesign. In: AUSTRALIAN COMPUTER SOCIETY, INC. *Proceedings of the Sixth Australasian conference on User interface-Volume 40*. [S.I.], 2005. p. 47–53. Citado na página 11.
- HAEFLIGER, S.; KROGH, G. V.; SPAETH, S. Code reuse in open source software. *Management Science*, INFORMS, v. 54, n. 1, p. 180–193, 2008. Citado na página 12.
- MYERS, B.; HUDSON, S. E.; PAUSCH, R. Past, present, and future of user interface software tools. *ACM Transactions on Computer-Human Interaction (TOCHI)*, ACM, v. 7, n. 1, p. 3–28, 2000. Citado na página 12.
- MYERS, B. A. Uimss, toolkits, interface builders1. 1996. Citado na página 12.
- POPPER, K. R. The logic of scientific discovery. Hutchinson, 1959. Citado na página 18.
- ROSENTHAL, D. A simple x11 client program—or how hard can it really be to write "hello, world"? In: USENIX Winter. [S.I.: s.n.], 1988. p. 229–242. Citado na página 12.
- SARTORI, R. Neurociência e comportamento na educação de criança e adolescentes. *Cadernos de Pesquisa*, 2010. Citado na página 11.
- SHNEIDERMAN, B. *Designing the user interface: strategies for effective human-computer interaction*. [S.I.]: Addison-Wesley Reading, MA, 1992. Citado na página 14.

THE GNOME PROJECT. *GNOME*. 2015. GNOME. Disponível em: <<https://gnome.org>>. Acesso em: 8 apr. 2015. Citado na página 13.

THE GTK+ PROJECT. *The GTK+ Project*. 2015. The GTK+ Project. Disponível em: <<http://gtk.org>>. Acesso em: 10 apr. 2015. Citado na página 14.

The Transmission Project. *Transmission*. Disponível em: <<http://transmissionbt.com>>. Citado na página 15.

THE TRANSMISSION PROJECT. *About Transmission*. 2015. About. Disponível em: <<http://www.transmissionbt.com/about/>>. Acesso em: 3 apr. 2015. Citado na página 15.

YAU, S. S.; COLLOFELLO, J. S.; MACGREGOR, T. Ripple effect analysis of software maintenance. In: *Computer Software and Applications Conference, 1978. COMPSAC'78. The IEEE Computer Society's Second International*. [S.l.: s.n.], 1978. p. 60–65. Citado na página 14.